INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume3, Issue2)

Feature and Processing Of Recognition of Characters, Words & Connecting Motions

Deepa .D
M.E. Student, Dept.
Maharaja Institute of Technology.
Coimbatore

R. Dharmalingam
Assistant Professor &HOD, ECE Dept
Maharaja Institute of Technology
Coimbatore.

Abstract: Recognition & Modeling of characters, words & connecting motions is accomplished based on six-degree-of-freedom hand motion data. We address air-writing on two levels: motion characters and motion words. Isolated air-writing characters can be recognized similar to motion gestures although with increased sophistication and variability. For motion word recognition in which letters are connected and superimposed in the same virtual box in space, we build statistical models for words by concatenating clustered ligature models and individual letter models. A hidden Markov model is used for air-writing modeling and recognition. We show that motion data along dimensions beyond a 2-D trajectory can be beneficially discriminative for air-writing recognition

***Keywords:** Air-Writing, Handwriting Recognition, Usability Study, 6-DOF Motion.*

I.INTRODUCTION

Motion gestures provide a complimentary modality for general human–computer interaction. Motion gestures are meant to be simple so that a user can easily memorize and perform them. However, motion gestures themselves are not expressive enough to input text for the motion-based control. We define “air-writing” as writing letters or words with hand or finger movements in a free space. Air-writing is especially useful for user interfaces that do not allow the user to type on a keyboard or write on a trackpad/touchscreen, or for text input for smart system control, among many applications.

In conventional handwriting, a sequential discrete stroke structure is made. A stroke is an isolated writing trajectory between the pen up/pen down events. In contrast, air-writing is rendered on a virtual plane without visual or haptic feedback and lacks the delimited sequence of writing events. Air-writing is also more complex for automatic recognition than cursive style writing on paper due to the lack of a concrete anchoring or reference position; the person who performs air-writing can only use an imaginary coordinate to guide the writing motion. The variability of motion data that represents a letter is thus considerably broader in air-writing than in paper writing. Isolated air-writing carries the assumption that the hand motion to render a letter has already been roughly localized in time and in space. Localization of motion rendering may be accomplished by use of a tracker, which can be easily turned ON or OFF, to signify the beginning and ending of a writing activity. The localization is only approximate and not fluctuation-free because most users cannot precisely synchronize the tracker control (ON–OFF) and the true writing trajectory. This is similar to the notorious problem of end-pointing in spoken

II. PRESENT WORK

We have to use this technology that will help to recognize the letters, characters, and words etc. on air. It uses IR Txs, IR Rxs, Demultiplexers, Multiplexers, and Microcontrollers for recognition purposes.

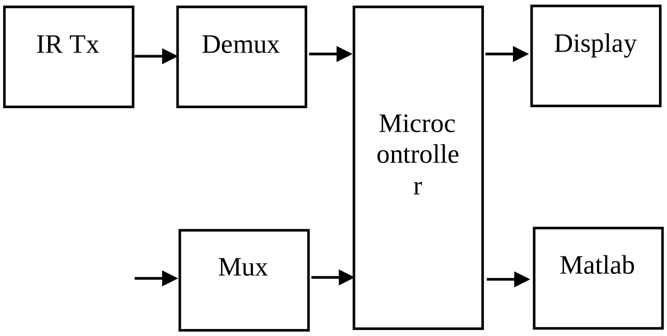


Fig 1 Block diagram of recognition of characters, motion

The computer mouse is one of the most common input devices on computers. However, there are many variations to the standard computer mouse. Computer mice can have a single button or many buttons along with a scroll wheel. Wireless mice are available. Mice also may have trackballs or use optical light to sense movement. However, while computer mice are because of several advantages, there are still a few disadvantages to computer mice as well. Computer mice need an unobstructed and flat surface to effectively monitor and manage user movements. However, flat surfaces may not always be available, especially as computer users become more mobile with laptop computers, and this may cause the limited use of the mouse.

The proposed system converts air into mouse thus we can avoid the mentioned problems in the existing system. In front of a computer monitor an array of sensors arranged to detect the touch on a thin layer of air. The output of sensors will be enough to get the location of the touch and the position values will be sent to the computer system and controlling mouse point through the MATLAB application.

111. RELATED WORK

There are other text input modalities in addition to typing and writing. One alternative approach is a mixture of typing and writing. In Quick writing a user swipes strokes on zones and subzones to input the associated characters. Two Stick applies a similar concept to the two joysticks on a gamepad. Swype allows a user to enter words on a soft keyboard by sliding from the first letter of a word to its last letter and uses a language model to guess the intended word. Similar to typing on a virtual keyboard, swiping strokes also requires the user's attention to the visual feedback while inputting text and is not eyes-free. Hidden Markov models (HMMs) are widely used for online handwriting recognition. In ligature models are proposed to address online recognition of cursive handwriting, in which successive letters are connected without explicit pen-up moves. Motion-based handwriting can also be considered in parallel to motion gestures or sign language. Motion gesture recognition has been studied with different types of motion tracking devices was achieved with inertial sensors attached to a glove. Then proposed a vision-based approach for finger-writing character Sign language is more sophisticated than motion gestures. Many sign language recognition systems use HMMs with various sensing technologies, such as data gloves and vision-based techniques.

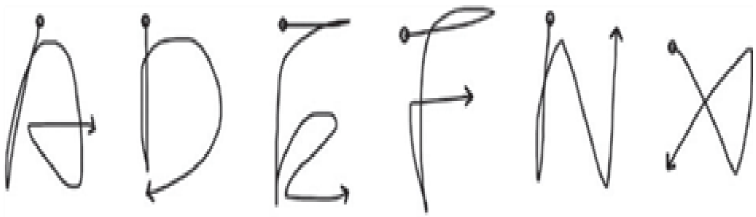
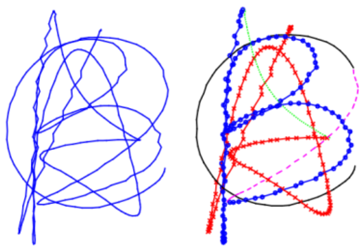


Fig. 2. Illustrations of the unit stroke writing of isolated letters. (a) A. (b) D. (c) E. (d) F. (e) N. (f) X.



(a) ABC (b) segmented ABC

Fig. 3 Two-dimensional projected trajectory of a motion word. (a) ABC. (b) Segmented ABC.

III.AIR-WRITING WITH SIX-DEGREE-OF-FREEDOM MOTION TRACKING

A. Unique Writing Style

Air-writing is fundamentally different from conventional handwriting on paper or a surface, which provides no haptic feedback. Similar to motion gestures, air-writing is tracked with a continuous stream of sensor data, and the writing is intuitively rendered in the air in uni-stroke without any pen-up and pen down information. The user envisions a writing box in the space and writes in this imaginary space without haptic feedback.

Fig. 2 and 3 demonstrate several key differences between air-writing (beyond single characters) and conventional handwriting. The first major difference is the lack of pen up/pen-down moves and haptic on feedback. The pen-up/pen-down information is useful for word segmentation of cursive handwriting and stroke delimitation for hand-print writing. The second is the box writing style with overlapping characters. Moreover, we track air-writing with 6-DOF motion data (translation and rotation), which is also different from the conventional 2- D spatial trajectory of pen-based writing. In our case, features derived from traditional handwriting recognition cannot be applied. Among related works of automatic air-writing recognition

B. Six-Degree-of-freedom Motion Tracking and Data Acquisition

We use a hybrid framework for 6-DOF motion tracking: the World viz PPT-X4 for optical tracking of the position of the infrared tracker and the Wii Remote Plus (Wii mote) for the inertial measurements of the acceleration and angular speed. The orientation is derived from a fusion of the acceleration and angular speed data. The system tracks a specially designed handheld device and provides both explicit (position and orientation) and implicit (acceleration and angular speed) 6-DOF data sampled at 60 Hz.

IV. AIR-WRITING PROCESSING AND MODELING

A. Feature Processing

From the 6-DOF motion data, we derive five features (observations): position *P* and velocity *V* from optical tracking, orientation *O*, acceleration *A*, and angular speed *W* from inertial tracking. Let $P^o = [p_x, p_y, p_z]^\top$ denote the positions, and $V^o = [\Delta p_x, \Delta p_y, \Delta p_z]^\top$ the rate of change in position. The orientation is represented in quaternion, $O^o = [q_w, q_x, q_y, q_z]^\top$.

TABLE I
DURATIONS (IN NUMBER OF SAMPLES) OF MOTION CHARACTERS BY 22 SUBJECTS

	avg	std		avg	std		avg	std
A	159.5	37.4	J	60.6	12.0	S	92.7	17.8
B	156.7	37.1	K	136.8	26.4	T	88.6	16.4
C	77.3	19.8	L	64.8	15.0	U	73.5	14.1
D	118.7	24.9	M	146.4	29.4	V	67.2	11.5
E	190.8	48.6	N	115.7	21.1	W	110.4	19.3
F	132.6	27.4	O	85.1	17.4	X	91.3	16.1
G	149.7	35.1	P	107.6	20.3	Y	105.7	20.5
H	137.5	29.9	Q	119.4	26.4	Z	94.1	18.6
I	42.8	10.3	R	134.9	24.5			

B. Air-Writing Modeling

HMM models for motion characters can be readily concatenated to form a motion word with additional connecting ligature motions. Such a modeling methodology is common, known as subword modeling, in large vocabulary continuous speech recognition to circumvent insufficient training data issues. Gesture recognition typically involves a limited vocabulary set. It is relatively easy to collect sufficient data of each gesture and straightforward to model each gesture directly from its own recordings. However, the vocabulary of air-writing can easily be thousands of words, and it is difficult to collect enough data for every word in the vocabulary. The data sufficiency problem prevents a designer from directly using whole "word" models.

The word-level HMM model is built upon these individual character models. A motion word is formed by connecting motion characters with ligature motions. Here, we define the ligature as the motion from the ending point of the preceding character to the starting point of the following character.

TABLE II CLUSTERS FOR START AND END POINTS OF CHARACTERS

Start point		Endpoint	
S1	BDEFHKLMNPRTUVWXYZ	E1	BDSX
S2	AIJOQ	E2	ITY
S3	CGS	E3	CEGHKLMQRZ
		E4	JP
		E5	AF
		E6	O
		E7	NUVW

In Table II, we manually cluster characters according to the position of the starting and the ending points. Based on the hard clustering, we generate several general questions, such as: *"Does the previous letter belong to* E1*?"* and *"Does the next letter belong to* S2*?"* To take into account both the endpoint position and the stroke direction, we further divide the hard clustering to create more detailed questions, e.g., *"is the previous letter* B *or* D*(ends at bottom left with a right-to-left)*

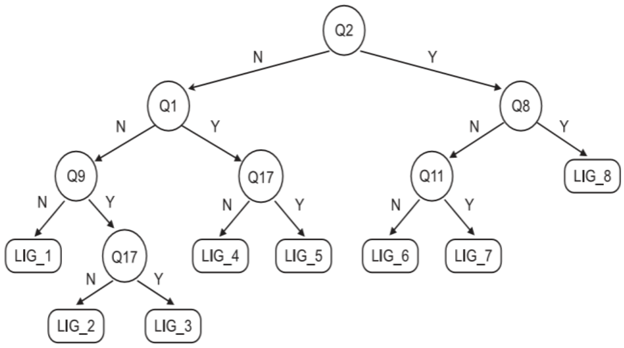


Fig. 4 Illustrative decision tree that results in eight clustered ligatures

Because the decision tree is data-driven, the questions and the resulting clusters vary upon the training data. During the branching process, not all questions in Table III are asked, and some questions may be asked multiple times. The real decision trees in our experiment have 30–40 branch nodes (asked questions) and 30–40 leaf nodes (clustered ligatures). With the help of the decision tree, we are able to synthesize unseen ligatures in the training data and generate models for all possible ligatures. The HMM for ABC now becomes A·lig_AB·B·lig_BC·C.

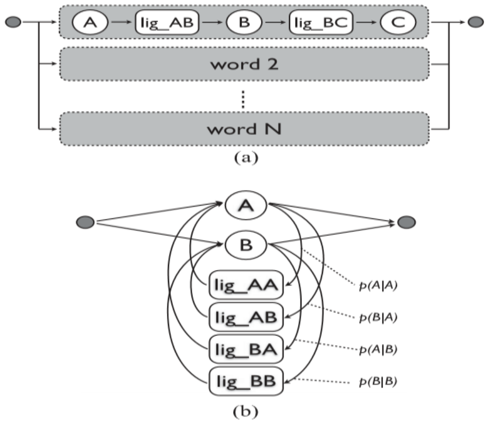


Fig. 5. Decoding word networks. (a) Word-based. (b) Letter-based (simplified)

V. MOTION CHARACTER AND MOTION WORD RECOGNITION EVALUATION

We first evaluate motion character recognition with the five basic features (*P*, *V* ,*O*, *A*,*W*) and different combinations of them, including *PV*, *AWO*, and *PV AWO*. The combinations of features actually correspond to different motion tracking devices. *PV* is the feature set derived purely from optical tracking, and *AWO* can be considered the full feature set from inertial measurements. *PV AWO* uses the available data from a hybrid 6-DOF motion tracking system.

A motion gesture can be defined in a 3-D space, but handwriting is actually defined on a 2-D surface regardless of the true writing motions. Therefore, we also investigate the feature $\hat{P}$ and $\hat{V}$, where $\hat{P}$ and $\hat{V}$ are the *x* and *y* components of *P* and *V*. We can consider $\hat{P}$ and $\hat{V}$ as representing the writing motion projected on a vertical plane.

TABLE 111

CER OF MOTION CHARACTER RECOGNITION

Features	CER (%) Average	std
P	3.72	(3.60)
V	6.12	(2.88)
O	3.81	(5.05)
A	7.97	(7.38)
W	7.92	(3.34)
PV	1.61	(2.16)
AWO	1.84	(2.37)
$PVOAW$	1.05	(1.23)
$\hat{P}$	3.88	(3.55)
$\hat{V}$	6.15	(2.69)
$\hat{P}\hat{V}$	1.61	(2.06)
$\hat{P}\hat{V}OAW$	1.05	(1.33)

A. Motion Character Recognition

The HMMs of motion characters are trained and tested with isolated characters, and we show the character error rate (CER) of leave-one-out cross-validation with different features in Table IV. First, we compare the discriminative power of the basic features. The explicit 6-D features (*P*, *V*, and *O*) outperform the implicit 6-D features (*A* and *W*).

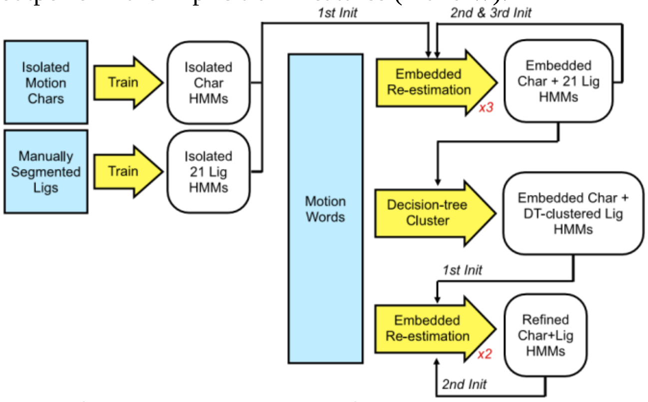


Fig. 5. Flow of the embedded re-estimation for character and ligature models

We can achieve robust motion character recognition even with AWO

B. Re estimated Character and Ligature Models

We obtain the HMMs of isolated motion characters and use them to initialize the character HMMs for motion word recognition. To construct the word model, we also need HMMs for ligatures. We propose two approaches to model ligatures: hard clustering and decision tree. First, we extract the ligatures from the manually segmented motion words in the 40-word vocabulary written by subject M1.the *AWO*. The CER of pure inertial tracking is slightly higher than the CER of pure optical tracking, and *PV AWO* achieves the lowest CER. The performance of motion character recognition supports our previous study on motion gesture recognition.

C. Word-Based Motion Word Recognition

For word-based word recognition, we use the refined HMMs of character and 21 hard clustered ligatures to build the decoding word network. The word-based word recognition is formulated as a one-out-of-*N* problem, where *N* is the vocabulary size. In the word-based decoding network, each path is a word model synthesized from corresponding character and ligature HMMs, and the letter sequences are tightly restricted to the vocabulary.

D. Letter-Based Motion Word Recognition

We choose the refined HMMs of character and decision-tree clustered ligatures to build the letter-based decoding word network. In our preliminary experiments of22 subjects and the 40-word vocabulary, ligature models clustered by a decision tree achieves about 2% absolute WER reduction over hard clustered ones. With the decision-tree-clustered ligatures as the only contextual constraint, the average WER of leave-one-out cross-validation of 22 subjects has a similar trend as the results of motion character recognition. To further improve the recognition performance, we utilize the statistics of letter sequences of the vocabulary. We estimate the bigram language model for the 40-word and 1k-word vocabulary separately.

TABLE 1V
USABILITY RESULT OF AIR-WRITING AND VIRTUAL KEYBOARD

	Air-writing			Virtual Keyboard		
Word Length	Time (Sec)	Distance (cm)	Attempt# Per Word	Time (Sec)	Distance (cm)	ExtraKey# Per Word
2	5.4	161	1.38	2.6	49	0.04
3	7.2	249	1.19	4.3	86	0.09
4	8.3	312	1.07	5.7	120	0.14
5	10.1	396	1.06	7.4	152	0.29
6+	14.0	566	1.04	9.2	174	0.29

RESULTS

We show the average writing/typing time and total traverse distance for words of different length in Table 1V. Because air writing is recognized on a word basis, we report the average number of attempts to correctly input a word. Longer words tend to have higher recognition accuracy and hence need fewer attempts. The average writing time of a two-letter word is 3.9 (= 5.4/1.38) s. For virtual keyboard, we report the average number of extra keystrokes, e.g., a typo and a backspace count as two extra keystrokes.

TABLE V
USABILITY RESULT OF AIR-WRITING AND VIRTUAL KEYBOARD

Question	air-handwriting	virtual keyboard
1. Intuitiveness [5: most intuitive]	4.10	4.75
2. Arm fatigue level [5: no fatigue]	3.05	3.10
3. Vote for inputting a short word (2-3 letters)	16	4
4. Vote for inputting a long word (4+ letters)	11	9
5. Satisfaction of recognition performance [5: most satisfied]	4.25	-

Air-writing is a variation of conventional writing, and virtual keyboard follows the same metaphor of typing on a touchscreen. Both methods are intuitive to users and have neutral scores for the arm fatigue level. Motions in the air involve more muscles than a keyboard or touch-based interaction and thus cause more fatigue. Even though the motion footprint of air-writing is three times larger, it does not directly reflect arm fatigue ratings. The arm fatigue level relates to the writing or typing style. For example, air writing could cause less fatigue for a user who rests the elbow and writes with the upper arm and wrist than a user who holds the whole arm in the air. The layout of the virtual keyboard is fixed for all subjects. To cover all keys, it requires a larger range of movement, e.g., the distance between key Z and Backspace is about 60 cm (1200 pixels). Six subjects mention that the keyboard layout is too big. Reducing the size of the keyboard layout can reduce the motion footprint. However, smaller keys can be prone to "typing" errors and require more precise pointing motions. The majority of users choose air-writing for short text input (two to three letters), and about half of users prefer air writing for long text input (4+ letters). There are other usability issues of air-writing from user feedback. The box-writing style appears to be easy to learn, but it needs some practice to write with the specified stroke order

CONCLUSION

We attempt to recognize air-writing with a 6DOF motion tracking system. The writing motion is tracked with the position and orientation in the global frame, and the acceleration and angular speed in the device-wise coordinates. The air-writing recording process is very time-consuming. To make the recording process feasible, we place constraints on stroke orders and uppercase letters with limited vocabulary to refine the scope of air-writing data acquisition without losing too much generality. From these motion data, we derive five basic features for observations of HMMs and from the combination of pure optical, pure inertial, and complete 6-DOF features. Although the handwriting is defined purely by the planar shape, we show that motion information beyond the spatial trajectory is informative for air-writing recognition.

Air-writing is uni-stroke without pen-up/pen-down information. The writing style and motor control are different from ordinary pen-based writing due to lack of haptic and vision feedback. We separate air-writing in two levels: motion characters and motion words. Motion characters are handled similar to motion gestures, and each character is modeled with an HMM. A motion word can be modeled by concatenating character and ligature models. We present two approaches to model ligatures: hard clustering and decision tree. The former to be sufficient for word-based word recognition. The latter provides the better capability of ligature modeling, which improves the performance of letter-based word recognition and stringent requirement on the accuracy. On the other hand, letter- The word-based word recognition achieves relatively low WER but is not able to recognize OOV words. The word-based recognizer is suitable for applications that have a limited vocabulary based word recognition has around 10% WER but can handle arbitrary letter sequences and progressive decoding.

REFERENCES

[1] M. Chen, G. AlRegib, and B.-H. Juang, "Air-writing recognition—Part II: Detection and recognition of writing activity in a continuous stream of motion data," *IEEE Trans. Human-Mach. Syst*, to be published.

[2] (2015). Leap motion. [Online]. Available: http://www.leapmotion.com

[3] M. Chen, G. AlRegib, and B.-H. Juang, "Feature processing and modeling for 6d motion gesture recognition," *IEEE Trans. Multimedia*, vol.15, no.3, pp. 561–571, Apr. 2013.

[4] C. H. Blickenstorfer, "Graffiti: Wow!" *Pen Computing Mag.*, vol. 1, pp. 30–31, Jan. 1995.

[5] K. Perlin, "Quikwriting: Continuous stylus-based text entry,"in*Proc.11th Ann. ACM Symp. User Interface Softw. Technol.*, 1998, pp. 215–216.

[6] P. Isokoski and R. Raisamo, "Quikwriting as a multi-device text entry method," in *Proc. 3rd Nordic Conf. Human-Computer Interaction*, 2004, pp. 105–108.

[7] T. Koltringer, ¨ P. Isokoski, and T. Grechenig, "Two stick: Writing with a game controller," in *Proc. Graph. Interface*, 2007, pp. 103–110.

[8] (2013). Swype—Type fast, swype faster. [Online]. Available: http://www. swype.com/

[9] R. Plamondon and S. Srihari, "Online and off-line handwriting recognition: a comprehensive survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 1, pp. 63 –84, Jan. 2000.

[10] J. Makhoul, T. Starner, R. Schwartz, and G. Chou, "On-line cursive handwriting recognition using hidden markov models and statistical grammars," in *Proc. Workshop Human Lang. Technol.*, 1994, pp. 432–43

